IN THE CLAIMS

Please amend the claims to the following.

1. (Currently Amended) A system for maintaining cache coherency in a CMP comprising:
an integrated circuit including
~~one or more~~ a plurality of processor cores, wherein the plurality of ~~one or more~~ processor cores each include a private cache;
a shared cache ~~separate from the plurality of cores~~ to be shared by the plurality of processor ~~one or more processor~~ cores, wherein the shared cache is to generate a first message to invalidate the block in a second processor core of the plurality of processor cores and provide a write acknowledgement to a requesting processor core, in response to receiving a write request referencing a block from the requesting processor core and the block not being owned; and
a ring to connect the plurality of ~~one or more~~ processor cores and the shared cache, the ring to transmit the first message to the requesting processor core and second processor core.

2. (Canceled)

3. (Currently Amended) The system of claim 1 wherein the shared cache includes one or more banks, wherein the one or more cache banks is responsible for a subset of a physical address space of the system, and wherein the block is associated with a physical address of the physical address space of the system.

4. (Currently Amended) The system of claim 1 [[3]] wherein the first message includes an InvalidateAndAcknowledge message , and wherein the shared cache is to generate the InvalidateAndAcknowledge message, further in response to the block being present in the shared cache and the second processor core being a custodian for the block. ~~the one or more cache banks is responsible for a subset of a physical address space of the system~~.

5. (Currently Amended) The system of claim 1 wherein the first message includes an InvalidateAllAndAcknowledge message , and wherein the shared cache is to generate the InvalidateAllAndAcknowledge message, further in response to the block not being present in the shared cache and none of the plurality of processor cores being a custodian for the block. the ~~one or more~~ processor cores are write-thru.

6. (Currently Amended) The system of claim [[5]] 1 wherein the plurality of ~~one or more~~ processor cores writes data through to the shared cache.

7. (Currently Amended) The system of claim 1 wherein the plurality of ~~one or more~~ processor cores each include[[s]] a merge buffer, and wherein each of the merge buffers are to coalesce multiple stores to a same block.

8. (Currently Amended) The system of claim 1 [[7]] wherein the shared cache is to fetch a second block from a memory and generate a write acknowledge message to provide a write acknowledgement to the requesting processor core in response to receiving a second write request referencing the second block, the second block not being present in the shared cache and not being owned by any of the plurality of processor cores ~~data is stored in the merge buffer~~.

9. (Currently Amended) The system of claim 8 wherein the shared cache is to generate an evict message to evict a third block from an owning processor core and generate a second write acknowledge message to provide a second write acknowledgment to the requesting processor core in response to receiving a third write request referencing the third block, the third block being present in the shared cache and the owning processor core of the plurality of cores owns the third block ~~the merger buffer purges data to the shared cache~~.

10. (Currently Amended) The system of claim 1 wherein a bank of the shared cache is to be a home location for a non-overlapping portion of a physical address space associated with the block. ~~the one or more processor cores accesses data from the shared cache~~.

11. (Currently Amended) The system of claim 7 [[8]] wherein each private cache of the plurality of cores are not to hold dirty data, and wherein each of the merger buffers are to hold the dirty data ~~coalesces multiple stores to a same block~~.

12. (Original) The system of claim 1 wherein the ring is a synchronous, unbuffered bidirectional ring interconnect.

13. (Currently Amended) The system of claim 12 wherein ~~a~~ the first message has a fixed deterministic latency around the ring interconnect.

14. (Currently Amended) An apparatus comprising:

an integrated circuit including: a plurality of cores and a shared memory connected in a ring, the shared memory ~~cache being separate from the plurality of cores, wherein each of the plurality of cores includes a private cache memory, and wherein the shared memory is~~ to be accessible by each of the plurality of cores, wherein each of the plurality of cores includes a private memory and a merge buffer to purge data to the shared memory, and wherein the shared memory is to generate an evict message referencing an address to an owning processor core of the plurality of cores in response to receiving a read request referencing the address from a requesting core of the plurality of cores and the owning processor core owning a block associated with the address.

15. (Currently Amended) The apparatus of claim 14, wherein the ~~plurality of cores and the shared memory are connected in a~~ ring includes ~~with~~ a synchronous unbuffered bi-directional ring interconnect.

16. (Previously Added) The apparatus of claim 14, wherein the shared memory is a shared cache including a plurality of blocks, and wherein the shared cache is capable of holding each of the plurality of blocks in a cache coherency state.

17. (Previously Added) The apparatus of claim 16, wherein the cache coherency state for each of the plurality of blocks is selected from a group consisting of (1) a not present state, (2) a present and owned by a core of the plurality of cores state, (3) a present, not owned, and custodian is a core of the plurality of cores state, and (4) a present, not owned, and no custodian state.

18. (Currently Amended) A[[n]] system comprising:

a processor including: a plurality of cores and a shared memory ~~separately~~ to be coupled together with an unbuffered bi-directional ring interconnect, wherein each of the plurality of cores is to be associated with a private cache memory, ~~and wherein~~ the shared memory is to be accessible by each of the plurality of cores, and the shared memory is to include a plurality of blocks, each of the plurality of blocks capable of being held in a not present state, a present and owned by a core of the plurality of cores state, a present, not owned, and a core of the plurality of cores is a custodian state, and a present, not owned, and no core of the plurality of cores is a custodian state; and

a system memory associated with the processor to hold elements to be stored by the shared memory.

19. (Currently Amended) The system ~~apparatus~~ of claim 18, wherein each of the plurality of blocks is a home location for a subset of a physical address space. ~~the shared memory is a shared cache including a plurality of blocks, and wherein the shared cache is capable of holding each of the plurality of blocks in a cache coherency state~~.

20. (Currently Amended) The system ~~apparatus~~ of claim 19, wherein the shared cache is to generate a first message to invalidate a requested block in all cores of the plurality of cores except for a requesting core of the plurality of cores, in response to receiving a write request referencing the requested block from the requesting core and requested block being held in the present, not owned, and no core of the plurality of cores is a custodian state. ~~the cache coherency state for each of the plurality of blocks is selected from a group consisting of (1) a not present state, (2) a present and owned by a core of the plurality of cores state, (3) a present, not owned, and custodian is a core of the plurality of cores state, and (4) a present, not owned, and no custodian state~~.